



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|-------------|----------------------|-------------------------------|------------------------|
| 10/807,678 | 03/24/2004 | Joseph Gil | YOR919990210US2 (8728-282) | 4311 |
| 46069 7590 06/18/2007 F. CHAU & ASSOCIATES, LLC 130 WOODBURY ROAD WOODBURY, NY 11797 | | | EXAMINER INGBERG, TODD D | |
| | | | ART UNIT 2193 | PAPER NUMBER |
| | | | MAIL DATE 06/18/2007 | DELIVERY MODE PAPER |

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

| | | | |
|------------------------------|--------------------------------------|-----------------------------------|--|
| Office Action Summary | Application No. 10/807,678 | Applicant(s) GIL ET AL. | |
| | Examiner Todd Ingberg | Art Unit 2193 | |

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 24 March 2004.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 21-41 is/are pending in the application.
- 4a) Of the above claim(s) 1-20 is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 21-41 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 24 March 2004 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau. (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date <u>3/24/04</u> . | 6) <input type="checkbox"/> Other: _____ |

Art Unit: 2193

DETAILED ACTION

Claims 21 – 41 have been examined.

Claims 1 – 20 have been canceled.

Specification

1. The abstract of the disclosure is objected to because the Abstract is more than 150 words.

Correction is required. See MPEP § 608.01(b).

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

3. Claims 21 – 41 are rejected under 35 U.S.C. 103(a) as being unpatentable over Inside the C++ Object Model, Stanley Lippman 1996, in view of Object-Oriented Analysis and Design, by James Martin 1992.

Examiner's Interpretations

A. The number of level of inheritance is not distinguishable. The process is the same regardless (third class, fourth class etc) see Lippman page 11.

B. The number of classes present can be the maximal independent set.

21. (New) A method for compiling object oriented code to produce an application having a reduced size, comprising: determining a virtual base class of the object oriented code virtually derived by a plurality of classes, each subobject of the plurality of classes comprising a virtual base pointer to a subobject of the virtual base class; infining the virtual base class into at least a first class of the plurality of classes; and compiling the object oriented code to produce the application.

Art Unit: 2193

Examiner's Rejection

C++ as taught by Lippman with the teaching of Martin teaches a method of compiling to produce an application having a reduced size (Martin cancel inheritance), comprising: determining a virtual base class of the object oriented code virtually derived by a plurality of classes (Lippman, pages 8 – 10 (Note: Multiple Inheritance on page 9) , each subobject of the plurality of classes (Lippman, page 10) comprising a virtual base pointer to a subobject of the virtual base class (Lippman, pages 95 - Lippman, page 98, #2 to page 101); inlining the virtual base class into at least a first class of the plurality of classes (Lippman, page 5 and 151 - 157) ; and compiling the object oriented code to produce the application (Lippman, compilation of C++ with canceled inheritance – intended use of the C++ compiler). Lippman teaches the internals of the C++ object model but does not explicitly teach the well known ability to cancel inheritance in C++. It is Martin who teaches canceling inheritance (Martin, page 269). Therefore, one of ordinary skill at the time of invention would have known to combine the teaching of Lippman and Martin, because canceling inheritance enable programs to be more efficient and better resemble the problem they are modeling.

22. (New) The method of claim 21, wherein inlining comprises: eliminating a virtual base pointer to the subobject of the virtual base class from a memory layout of the first class; and storing the subobject of the virtual base class in a fixed offset in the memory layout of the first class in which the virtual base pointer was eliminated.

Examiner's Rejection

See Lippman, pages 26-28

23. (New) The method of claim 21, further comprising: determining the first class derived from the virtual base class to be duplicated in a second class that inherits from the first class; and preventing inlining of the virtual base class into the first class.

Examiner's Rejection

As per claim 21 canceling inheritance prevents inlining.

24. (New) The method of claim 21, further comprising: determining a third class that inherits from the virtual base class through a set of classes, where each class of the set of classes virtually inherits from the virtual base class; and inlining the virtual base class into at least a second class of the set of classes.

Examiner's Rejection

Inlining as per claim 22.

25. (New) The method of claim 24, wherein the set of classes is a maximal independent set.

Examiner's Rejection

The result of inheritance produces a maximal independent set (As per claim 21).

26. (New) The method of claim 24, where inlining comprises: eliminating a virtual base pointer to the subobject of the virtual base class from the memory layout of the second class; and storing the subobject of the virtual base class in a fixed offset in the memory layout of the second class in which the virtual base pointer was eliminated.

Art Unit: 2193

Examiner's Rejection

The mechanism as per claim 22. The level of inheritance in the model is not a factor in determining patentability. It does indicate more than one level which is well known to be part of C++ as per claim 21.

27. (New) The method of claim 24, further comprising: determining the second class derived from the virtual base to be duplicated in a fourth class that inherits from the second class; and preventing inlining of the virtual base class into the second class.

Examiner's Rejection

As per the rejection for claim 26.

28. (New) A method for compiling object oriented code to produce an application having a reduced size, comprising: removing transitive virtual inheritance from between a virtual base class and a first class inheriting from the virtual base class, wherein the first class inherits non-virtually from a second class and the second class inherits virtually from the virtual base class, upon determining the presence of transitive virtual inheritance; removing single virtual inheritance from between the virtual base class and a third class virtually inheriting from the virtual base class, wherein the third class is the only class inheriting from the virtual base class, upon determining the presence of single virtual inheritance; determining a virtual base class of the object oriented code virtually derived by a plurality of classes, each subobject of the plurality of classes comprising a virtual base pointer to a subobject of the virtual base class; inlining the virtual base class into at least a fourth class of a plurality of classes virtually derived from the virtual base class; and compiling the object oriented code to produce the application.

See the rejection for claim 21.

29. (New) The method of claim 28, wherein inlining comprises: eliminating a virtual base pointer to a subobject of the virtual base class from a memory layout of the fourth class; and storing the subobject of the virtual base class in a fixed offset in the memory layout of the fourth class in which the virtual base pointer was eliminated.

See the rejection for claim 22.

30. (New) The method of claim 28, further comprising: determining the fourth class derived from the virtual base class to be duplicated in a fifth class that inherits from the fourth class; and preventing inlining of the virtual base class into the fourth class.

See the rejection for claim 23.

31. (New) The method of claim 28, further comprising: determining a seventh class that inherits from the virtual base class through a set of classes, where each class of the set of classes virtually inherits from the virtual base class; and inlining the virtual base class into at least a sixth class of the plurality of classes that virtually inherit from the virtual base class.

See the rejection for claim 24.

32. (New) The method of claim 31, wherein the set of classes is a maximal independent set.

Art Unit: 2193

See the rejection for claim 25.

33. (New) The method of claim 31, wherein inlining comprises: eliminating a virtual base pointer to the subobject of the virtual base class from the memory layout of the sixth class; and storing the subobject of the virtual base class in a fixed offset in the memory layout of the sixth class in which the virtual base pointer was eliminated.

See the rejection for claim 26.

34. (New) The method of claim 31, further comprising: determining the sixth class derived from the virtual base class to be duplicated in a eighth class that inherits from the sixth class; and preventing inlining of the virtual base class into the sixth class.

See the rejection for claim 27.

35. (New) A method for compiling object oriented code to produce an application having a reduced size, comprising: removing transitive virtual inheritance from between a virtual base class and a first class inheriting from the virtual base class, wherein the first class inherits non-virtually from a second class and the second class inherits virtually from the virtual base class, upon determining the presence of transitive virtual inheritance; determining a virtual base class of the object oriented code virtually derived by a plurality of classes, each subobject of the plurality of classes comprising a virtual base pointer to a subobject of the virtual base class; inlining the virtual base class into at least a third class of a plurality of classes virtually derived from the virtual base class; and compiling the object oriented code to produce the application.

See the rejection for claim 21.

36. (New) The method of claim 35, wherein inlining comprises: eliminating a virtual base pointer to the subobject of the virtual base class from the memory layout of the third class; and storing the subobject of the virtual base class in a fixed offset in the memory layout of the third class in which the virtual base pointer was eliminated.

See the rejection for claim 22.

37. (New) The method of claim 35, further comprising: determining the third class derived from the virtual base class to be duplicated in a fourth class that inherits from the third class; and preventing inlining of the virtual base class into the third class.

See the rejection for claim 23.

38. (New) The method of claim 35, further comprising: determining a fourth class that inherits from the virtual base class through a set of classes, where each class of the set of classes virtually inherits from the virtual base class; and

inlining the virtual base class into at least a fifth class of the plurality of classes that virtually inherit from the virtual base class.

See the rejection for claim 24.

39. (New) The method of claim 38, wherein the set of classes is a maximal independent set.

See the rejection for claim 25.

Art Unit: 2193

40. (New) The method of claim 38, wherein inlining comprises: eliminating a virtual base pointer to the subobject of the virtual base class from a memory layout of the fifth class; and storing the subobject of the virtual base class in a fixed offset in the memory layout of the fifth class in which the virtual base pointer was eliminated.

See the rejection for claim 26.

41. (New) The method of claim 38, further comprising: determining the fifth class derived from the virtual base class to be duplicated in a sixth class that inherits from the fifth class; and preventing inlining of the virtual base class into the fifth class.

See the rejection for claim 27.

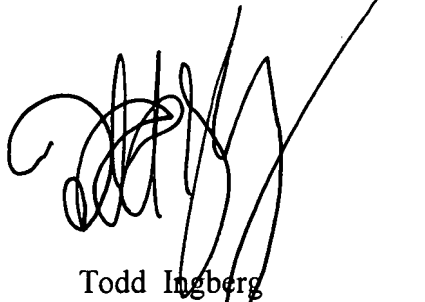
Correspondence Information

4. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Todd Ingberg whose telephone number is (571) 272-3723. The examiner can normally be reached on during the work week..

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2193

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.



Todd Ingberg
Primary Examiner
Art Unit 2193

TI